



# UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE  
United States Patent and Trademark Office  
Address: COMMISSIONER FOR PATENTS  
P.O. Box 1450  
Alexandria, Virginia 22313-1450  
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/811,305	03/29/2004	Michael John Wray	200300134-2	8275

22879 7590 09/03/2009

HEWLETT-PACKARD COMPANY  
Intellectual Property Administration  
3404 E. Harmony Road  
Mail Stop 35  
FORT COLLINS, CO 80528

EXAMINER
----------

WANG, HARRIS C

ART UNIT	PAPER NUMBER
----------	--------------

2439

NOTIFICATION DATE	DELIVERY MODE
-------------------	---------------

09/03/2009

ELECTRONIC

**Please find below and/or attached an Office communication concerning this application or proceeding.**

The time period for reply, if any, is set in the attached communication.

Notice of the Office communication was sent electronically on above-indicated "Notification Date" to the following e-mail address(es):

JERRY.SHORMA@HP.COM  
ipa.mail@hp.com  
jessica.l.fusek@hp.com



UNITED STATES PATENT AND TRADEMARK OFFICE

---

Commissioner for Patents  
United States Patent and Trademark Office  
P.O. Box 1450  
Alexandria, VA 22313-1450  
[www.uspto.gov](http://www.uspto.gov)

**BEFORE THE BOARD OF PATENT APPEALS  
AND INTERFERENCES**

Application Number: 10/811,305  
Filing Date: March 29, 2004  
Appellant(s): WRAY ET AL.

---

Ashkok K. Mannava  
For Appellant

**EXAMINER'S ANSWER**

This is in response to the appeal brief filed 5/19/2009 appealing from the Office action mailed 1/16/2009.

**(1) Real Party in Interest**

A statement identifying by name the real party in interest is contained in the brief.

**(2) Related Appeals and Interferences**

The examiner is not aware of any related appeals, interferences, or judicial proceedings which will directly affect or be directly affected by or have a bearing on the Board's decision in the pending appeal.

**(3) Status of Claims**

The statement of the status of claims contained in the brief is correct.

**(4) Status of Amendments After Final**

The appellant's statement of the status of amendments after final rejection contained in the brief is correct.

**(5) Summary of Claimed Subject Matter**

The summary of claimed subject matter contained in the brief is correct.

**(6) Grounds of Rejection to be Reviewed on Appeal**

The appellant's statement of the grounds of rejection to be reviewed on appeal is correct

**(7) Claims Appendix**

.The copy of the appealed claims contained in the Appendix to the brief is correct.

**(8) Evidence Relied Upon**

2002/0194496	GRIFFIN	12-2002
2004/0003288	WISEMAN	1-2004

**(9) Grounds of Rejection**

The following ground(s) of rejection are applicable to the appealed claims:

**Claims 1-2, 10-12 are rejected under 35 U.S.C. 102(b) as being anticipated by Griffin.**

Regarding Claim 1, 12

Griffin (US 20020194496) teaches a system comprising a trusted computing platform including:

at least one first logically protected computing compartment associated with initialization of said system and

at least one second logically protected computing compartment, second logically computing compartment being associated with at least one service or process supported by said system, (*"Each resource of the computing platform which it is desired to protect is given a label indicating the compartment to which that resource belongs. Mandatory access controls are performed by the kernel of the host operating system to ensure that resources from one compartment cannot interfere with resources from another compartment."*

*Paragraph [0034])*

wherein the system is arranged to load onto said trusted computing platform a predetermined security policy including at least one security rule for controlling the operation of each of said logically protected computing compartments ("The actions or privileges within a component are constrained, particularly to restrict the ability of a process to execute methods and operations which have effect outside the compartment" Paragraph [0031]) The Examiner interprets the at least one security rule as each of the methods of each compartments are restricted within itself.;

wherein the security rule relating to the at least one first logically protected computing compartment is arranged to be loaded onto said trusted computing platform when the system is initialized (*"the trusted device performs a secure boot process when the computing platform is reset to ensure that the host operating system of the platform is running properly and in a secure manner"* Paragraph [0025]) and

wherein the at least one security rule relating to the at least one second logically protected computing compartment is only arranged to be loaded onto said trusted computing platform if one or more services or processes associated therewith are enabled (*"access control checks are performed such as through the use of hooks to a dynamically loaded security module that consults a table of rules indicating which compartments are allowed access the resources of another compartment. In the absence of a rule explicitly allowing a cross compartment access to take place, an access attempt is denied by the kernel"* Paragraph [0036])(*"Multiple applications can be run on the guest operating system, each within a separate compartment of the guest operating system. This embodiment enables each computing environment to be subdivided"* Paragraph[0068])

Art Unit: 2439

Regarding Claim 2,

Griffin teaches a system according to claim 1, wherein one or more common variable is defined for each compartment, wherein a relevant security rule is only arranged to be added if the variable associated with a particular compartment is enabled (*"Each resource of the computing platform which it is desired to protect is given a label indicating the compartment to which that resource belongs. Mandatory access controls are performed by the kernel of the host operating system to ensure that resources from one compartment cannot interfere with resources from another compartment. Access controls can follow relatively simple rules, such as requiring an exact match of the label" Paragraph [0034])*

Regarding Claim 10,

Griffin teaches a system according to claim 1, wherein the at least one compartment includes an operating system arranged to be controlled by the operating system kernel(*"the compartment is an operating system compartment controlled by a kernel of the host operating system" Paragraph[0032] of Griffin*)

Regarding Claim 11,

Griffin teaches the system according to claim 1, including means for determining when a service is starting, and on being enabled, for loading the compartment associated with that service and loading the security rules associated with that service (*"access control*

Art Unit: 2439

*checks are performed such as through the use of hooks to a dynamically loaded security module that consults a table of rules indicating which compartments are allowed access the resources of another compartment. In the absence of a rule explicitly allowing a cross compartment access to take place, an access attempt is denied by the kernel" Paragraph [0036]) The Examiner interprets an access attempt for the service as "a service starting."*

**Claims 3-9 are rejected under 35 U.S.C. 103(a) as being unpatentable over Griffin in view of Wiseman.**

Regarding Claims 3-4,

Griffin teaches the system according to claim 2. Griffin does not explicitly teach wherein at least one variable associated with a directory of plug-ins is arranged to be added wherein the system is arranged to determine, in response to a compartment being enabled, a status of said at least one variable and cause a relevant plug-in based upon a directory of plug-ins to run only if an associated variable is 'true'

Wiseman (20040003288) teaches at least one variable associated with plug-ins is arranged to be added wherein the system is arranged to determine in response to a compartment being enabled (*The Main Platform Initialization Code performs necessary functions to complete the initialization of the platform. Such functions may include initializing devices embedded within the platform, and locating and initializing optional plug-in or*

Art Unit: 2439

*embedded adapters (having their own device initialization code). After this, the Main Platform Initialization Code locates the OS Loader and executes it. The OS Loader, in turn, loads the OS into memory and begins executing the OS. At this point, the platform is considered in the OS-present state and is fully under control of the loaded OS)*

. It would have been obvious to one of ordinary skill in the art at the time of the invention to modify Griffin to accept plugins during initialization as taught by Wiseman.

The motivation is to extend the capabilities of the services.

Regarding claim 5,

Griffin and Wiseman teach a system according to claim 4, wherein the at least one compartment includes an operating system compartment arranged to be controlled by the operating system kernel (*"the compartment is an operating system compartment controlled by a kernel of the host operating system" Paragraph[0032] of Griffin*)

Regarding Claim 6,

Griffin and Wiseman teach a system according to claim 5, wherein the at least one compartment and network resources are arranged so communication between them is provided via relatively narrow kernel level controlled interfaces to a transport mechanism (*"Communication between compartments is provided using narrow kernel level controlled interfaces to a transport mechanism such as TCP/UDP" Paragraph[0036] of Griffin*)



Art Unit: 2439

Regarding Claim 7,

Griffin and Wiseman teach a system according to claim 6, wherein said communication is governed by rules specified on a compartment by compartment basis (*"Access to these communication interfaces is governed by rules specified on a compartment by compartment basis" Paragraph [0036] of Griffin*)

Regarding Claims 8-9,

Griffin and Wiseman teach a system according to claim 7, including means for determining when a service is starting, and on being enabled, for loading the compartment associated with that service and loading the at least one security rule associated with that service (*"access control checks are performed such as through the use of hooks to a dynamically loaded security module that consults a table of rules indicating which compartments are allowed access the resources of another compartment. In the absence of a rule explicitly allowing a cross compartment access to take place, an access attempt is denied by the kernel" Paragraph [0036]) The Examiner interprets an access attempt for the service as "a service starting."*

### **(10) Response to Argument**

Appellant's propose two arguments. First Appellant argues that Griffin does not anticipate every element in the claimed invention. Second the Appellant argues claims

Art Unit: 2439

3-9 recite additional features not found in the cited documents of record. Examiner respectfully disagrees for the reasons set forth below.

I.

Appellant argues that “claim 1 recites different timings for loading security rules: (1) the security rule for the first compartment is arranged to be loaded **when** the system is initialized and (2) the security rule for the second compartment is arranged to be loaded **when** the services or processes associated with the services or processes associated with the second compartment are enabled (pg. 7 of Appeal Brief, emphasis added by the Examiner)

At the outset, the Examiner disagrees with the above argument. The claim language requires that (1) the security rule for the first compartment is arranged **when** the system is initialized and (2) the security rule for the second compartment is arranged to be loaded **if** the services or processes associated with the services or processes associated with the second compartment are enabled.

More specifically, the Appellant is arguing in the second security rule is differentiated from Griffin because “the security rules in Griffin are loaded at the time the computing platform is reset, but not at different times when the services or processes are enabled, as recited in claim 1 (pg. 8 of Appeal Brief, emphasis added).”

However the Appellant’s arguments are not consistent with the claim language. There is a distinct difference between loading a security rule when the services are enabled, as argued, as opposed to loading a security rule if the services are enabled as

Art Unit: 2439

claimed. As claimed, the timing of the loading of the second security rule is not mentioned, only an "if" condition. As such, arguments regarding any timing of the loading of the rules in the second compartment (See lines 17-23 of pg. 8 of Appeal Brief) are considered spurious.

Appellant argues that "Griffin fails to teach that a security rule for an application is loaded only if the service for the application is enabled (pg. 9)."

The Examiner disagrees. The cited paragraph of Griffin (Paragraph [0036]) teaches "a dynamically loadable security module that consults a table of rules indicating which compartments are allowed to access resources of another compartment. In the absence of a rule explicitly allowing a cross compartment access to take place, an access attempt is denied by the kernel."

Therefore Griffin teaches a security rule which is loaded only if communication or access between compartments is enabled.

The Applicant argues the rejection of dependent claim 2 fails to teach "that a relevant security rule for a compartment is only arranged to be added if a common variable for that particular compartment is enabled (pg. 10 of Appeal Brief)."

Paragraph [0034] teaches rules being enforced when the resources (the common variable) are enabled.

II.

The Appellant argues that "claims 3-9 recite additional features not found in the cited documents of record...For instance, claim 4 recites that the system is arranged to cause a relevant plug-in to run if an associated variable is true...Wiseman also fails to mention any plug-ins being operated when a variable is true (pg. 13 of Appeal Brief)."

The Examiner respectfully disagrees. The cited paragraph of Wiseman recites "The Main Platform Initialization Code performs necessary functions to complete the initialization of the platform. Such functions may include the initializing devices embedded within the platform, and locating and initializing optional plug-in or embedded adaptors (Paragraph [0005])."

The Examiner interprets the determining of whether there are necessary functions to complete initialization of the platform to be the "variable" recited in Claims 3-4. If there are no necessary functions to complete initialization (the "false" condition), the OS loader is executed. However if there are necessary functions to complete initialization (the "true" condition) plug-ins are located and initialized. The Examiner equates this to the claim language "caus[ing] a relevant plug-in based upon the directory of plug-ins to run only if an associated variable is "true." Therefore Wiseman teaches plug-ins being operated when a variable is "true."

#### **(11) Related Proceeding(s) Appendix**

No decision rendered by a court or the Board is identified by the examiner in the Related Appeals and Interferences section of this examiner's answer.

Art Unit: 2439

For the above reasons, it is believed that the rejections should be sustained.

Respectfully submitted,

/Harris C Wang/

Examiner, Art Unit 2439

Conferees:

Christopher Brown

/Christopher J Brown/

Primary Examiner, Art Unit 2439

Edan Orgad

/Edan Orgad/

Supervisory Patent Examiner, Art Unit 2439